A Novel Verifiable Dynamic Multi-Policy Secret Sharing Scheme

Feng Wang [1,2,3], Lize Gu [1,2,3], Shihui Zheng [1,2,3], Yixian Yang [1,2,3], Zhengming Hu [1,2,3]

1 *Information Security Center, State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China*

2 Key Laboratory of Network and Information Attack and Defence Technology of MOE Beijing University of Posts and Telecommunications, Beijing 100876, China

3 *National Engineering Laboratory for Disaster Backup and Recovery, Beijing 100876, China Beijing University of Posts and Telecommunications, Beijing 100876, China*

flys99@126.com, glz_bupt@263.net

Abstract: **Most verifiable multi-secret sharing schemes can not distinguish the security classification because of having the common drawback of sharing secrets with the same threshold value. In order to withstand the problem, a novel verifiable dynamic multi-policy secret sharing scheme is proposed, in which the dealer can freely give any sets of group secrets for sharing according to the different threshold values, and the shadows held by the participants are not only reusable but also verifiable. The proposed scheme provides efficient solutions against the malicious dealer and participants. Moreover, the proposed scheme can add new participants and new group secrets. Compared to the current secret sharing scheme, the proposed scheme is more flexible and more practical application.**

Keywords: **verifiable, dynamic, multi-policy, key management, threshold**

I. INTRODUCTION

Secret sharing schemes are important techniques for key management [1, 2]. The first secret sharing schemes were proposed by Shamir[3] and Blackly[4] respectively in 1979. Shamir's scheme was based on the Lagrange interpolating polynomial, while Blakley's scheme was based on linear projective geometry. In their schemes, the dealer splits the secret into n different pieces called shadows, which are given to the participants over a secret channel. At least t or more participants can pool their shadows and easily reconstruct the secret, but only $t-1$ or fewer secret shadows cannot. As the intensive study, the secret sharing schemes of Shamir and Blakley can't solve the following problem:

(1) Only one secret can be shared during one secret sharing process, their schemes are not multi-secret sharing schemes [5].
(2) Once the secret has been reconstructed, it is required that the dealer redistributes a fresh shadow over a secret channel to every participant. The shadows of participants are not reusable in their schemes [6].
(3) A dishonest dealer may distribute a fake shadow to a certain participant, and then that participant would subsequently never obtain the true secret, their schemes can't detect dishonest dealer [7].
(4) A malicious participant may provide a fake shadow to other participants, which makes the malicious participant the only one who can reconstruct the true secret. Their schemes can not resist the deceive behaviors of dishonest participants [8].
(5) The dealer can't add new secrets to the system. [9]
(6) The dealer can't distribute a shadow to a new participant who is a new member of the group. [10]
(7) The dealer can't share the group secrets corresponding to the different threshold values. [11, 12]

To solve the above problems, a lot of researchers make a great effort in multi-secret sharing schemes and verifiable secret sharing schemes [13, 14, 15, 16]. To overcome the drawback in (1), some efficient (t,n) multi-secret sharing schemes have been proposed to share multiple secrets [17, 18, 19, 20, 21]. To deal with the drawback in (2), Jackson et al. [17] have further proposed multi-use scheme. The difference is that the shadow kept by each participant in a multi-use scheme is reusable after secret reconstruction. So the scheme of Jackson et al can avoid the cost of time and resources to redistribute shadows. However, their schemes still have the common drawbacks in (3) and (4).

To do away with the drawback in (3), Chor et al. [22] have proposed a verifiable secret sharing (VSS) scheme to detect cheating by a dishonest dealer. However, the problem in (4) still exists in their scheme. In order to solve the problem in (4), Stadler [23] proposed a VSS scheme which is not only robust against the cheating by the dealer but also against the cheating by any participants. Nevertheless, both VSS schemes can only deal with one secret in one secret sharing process.

Taking all the above problems in (1), (2), (3), (4) into consideration, Harn[18] has proposed a (t,n) threshold verifiable multi-secret sharing (VMSS) scheme which can detect malicious dealer and dishonest participants. In Harn's scheme, every participant keeps only one reusable shadow to share any set of secrets. However, Lin and Wu [19] pointed out that Harn's scheme suffered from the problems of large $n!/((n-t)!\cdot t!)$ module exponentiations and running

interactive verification protocol to verify each other. Except that Harn's scheme still exits the problems in (5), (6), (7). Chen et al. [20] have proposed an alternative (t,n) VSS scheme to avoid the disadvantages in Harn's scheme [18]. However, Lin and Wu [19] have also pointed out that Chen et al.'s scheme is inefficient because the dealer has to record all participants' shadows and take $2n$ modulo exponentiations to compute an n-dimensional verification vector for each shard secret. In order to avoid the disadvantages in Harn's scheme and problem in (5) and improve the performance of Chen et al.'s scheme, Lin and Wu proposed a (t,n) threshold VMSS scheme based on the intractability of factorization and the problem of discrete logarithm module a composite[19]. However, He and Wu [21] have indicated that a malicious participant can provide a fake sub-shadow to cheat other honest participants. To withstand the problem in Lin and Wu's scheme, Chang et al. [9] proposed an improvement on the Lin-Wu (t,n) threshold VMSS scheme. However, Chang et al.'s [9] scheme still exits the problem in (6) and (7). In 2007, Geng et al. [11] proposed a multi-policy secret sharing scheme to overcome the problem in (7). In their scheme, different secrets can be reconstructed corresponding to different threshold values. In 2008, Lin et al. [12] also proposed a multi-policy secret sharing scheme. Compared to Geng et al.'s scheme, Lin et al.'s scheme is more efficient. However, both of their schemes can't solve the problems in (3), (4), (5) and (6).

Taking all the above problems into consideration, a verifiable dynamic multi-policy secret sharing scheme is proposed. The proposed scheme will have the following features:

1) The dealer can arbitrarily give any set of group secrets for sharing, and only one shadow, which is reusable, should be kept by each participant. This solves the problems in (1), (2).
2) Every participant can detect any cheating by the dealer and verify his/her own shadow. This solves the problem in (3).
3) Every participant can detect the cheating by any other participants by using a non-interactive verification protocol and verify his/her sub-shadow. This solves the problem in (4).
4) The dealer can dynamically add new participants and new secrets to the system. This solves the problems in (5), (6).
5) Group secrets can be shared according to different threshold values and every threshold value corresponding many different group secrets. This solves the problem in (7).

The remainder of our paper is organized as follows. In Section II, we shall propose our verifiable dynamic multi-policy threshold secret sharing scheme. In Section III, we shall give the analysis of the proposed scheme. Finally, our conclusion will be in Section IV.

II. VERIFIABLE DYNAMIC MULTI-POLICY THRESHOLD SECRET SHARING SCHEME

In this section, we will propose a novel verifiable dynamic multi-policy threshold secret sharing scheme. The most property of the scheme is that the group secrets can be shared according to different threshold values. This has an important practical significance. Such as the secret of vault should not have the same threshold value as the doors' of the bank. Our scheme is compromised four phases: *A*. initialization stage, *B*. shadow generation and verification stage, *C*. credit ticket generation stage, *D*. sub-shadow verification and secret reconstruction stage. The details of four stages are as follows:

A. Initialization stage

In this stage, the dealer (denoted as $\mathrm{U_D}$) first creates a public notice board (NB) which is used for storing necessary public parameters. The participants can access those parameters on the NB. The contents on the board can only be modified or updated by $\mathrm{U_D}$.

p, q: tow large prime numbers satisfying the security of RSA public key cryptography[24], and $p = 2p' + 1$, $q = 2q' + 1$ (p' and q' are also large prime numbers);

N: $N = pq$;

$\varphi(N)$: $\varphi(N)$ is Euler's function, $\varphi(N) = (p-1)\cdot(q-1)$

R: $R = p'q'$;

g: a generator with order R of Z_N;

G: set of participants $G = \{\mathrm{U}_1, \mathrm{U}_2, \cdots, \mathrm{U}_n\}$, and the identity of U_j is $ID_j (j = 1,2,\cdots,n)$.

S: set of group secrets. $S = \{S_{i1}, S_{i2}, \cdots, S_{ik}\}$. $i \le n, k = 1,2,\cdots$

$h(\cdot)$: a secure one-way hash function which accepts input of any length and generates a fixed length output.

$\mathrm{U_D}$ puts $\{N, g\}$ on NB and keeps $\{R, \varphi(N)\}$ secret.

B. Shadow generation and verification stage

$\mathrm{U_D}$ randomly chooses $x_j (j = 1,2,\cdots,n)$ from Z_R as the participant U_j's shadow, then generates polynomials

$$f_i(x) = a_i + d_1 x + d_2 x^2 + \cdots + d_{i-1} x^{i-1} \bmod R$$

$(i = 1,2,\cdots,n)$

and computes:

$$C_{ij} = f_i(ID_j) \bmod R (i = 1,2,\cdots,n;\ j = 1,2,\cdots,n)$$

$$Y_{ij} = g^{C_{ij}} \bmod N (i = 1,2,\cdots,n;\ j = 1,2,\cdots,n)$$

$$K_{ij} = x_j \oplus C_{ij} \quad (i = 1,2,\cdots,n;\ j = 1,2,\cdots,n)$$

$\mathrm{U_D}$ publics $\{K_{ij}, Y_{ij}, g^{a_i}, g^{d_1}, \cdots, g^{d_{i-1}}\}$ $(i = 1,2,\cdots,n;\ j = 1,2,\cdots,n)$ on NB, and sends $x_j (j = 1,2,\cdots,n)$ to U_j through secure channel. When participant U_j receives x_j, he/she gets $\{K_{ij},\ g^{a_i}, g^{d_1}, \cdots, g^{d_{i-1}} (i = 1,2,\cdots,n\}$ from NB and computes:

$$x_{ij} = x_j \oplus K_{ij} \quad (i = 1,2,\cdots,n)$$

Then he/she can check the following equation to verify the validity of x_j.

$$g^{x_{ij}} = g^{a_i} \prod_{m=1}^{i-1} (g^{d_m})^{(ID_i)^m} \bmod N \qquad (1)$$

$(i = 1, 2, \cdots, n)$

If Eq.（1）does not hold, the shadow x_j distributed by U_D is not valid.

C. Credit ticket generation stage

In this phase, U_D performs the following steps to compute credit tickets $T_{i1}, T_{i2}, \cdots, T_{ik}$ $1 \le i \le n, k = 1, 2, \cdots$ for each group secret $S_{i1}, S_{i2}, \cdots, S_{ik} \in S$ $1 \le i \le n, k = 1, 2, \cdots$, and $S_{ik} (1 \le i \le n, k = 1, 2, \cdots)$ corresponding the threshold value $i (1 \le i \le n)$.

Step 1 U_D randomly chooses different public secret $e_{ik} \in [1, \varphi(N)]$ $1 \le i \le n$, $k = 1, 2, \cdots$ of RSA algorithm [24], and computes d_{ik} through the equation $e_{ik} \cdot d_{ik} = 1 \bmod \varphi(N)$

Step 2 U_D computes a credible ticket T_{ik} and a value H_{ik}

$T_{ik} = g^{d_{ik}} \bmod N$

$H_{ik} = S_{ik} \oplus h(g^{a_i d_{ik}}) \bmod N$ for $1 \le i \le n, k = 1, 2, \cdots$

Step 3 U_D puts the 3-tuple $\{e_{ik}, T_{ik}, H_{ik}\}$ $(1 \le i \le n, k = 1, 2, \cdots)$ on the NB.

D. Sub-shadow verification and secret reconstruction stage

Assumed reconstructing group secret S_{lm} with the threshold value l. Let $W(|W| = l)$ be any subset of l participants in G. Without loss of generality, assumed l participants $U_j \in W$ cooperate to reconstruct $S_{lm} \in S$. Every $U_j \in W$ performs the following the steps:

Step 1 Every $U_j \in W$ gets the 2-tuple (T_{lm}, K_{lj}) from the NB and uses his/her shadow x_j to compute a sub-shadow A_{lmj}:

$B_{lj} = K_{lj} \oplus x_j$

$A_{lmj} = (T_{lm})^{B_{lj}} \bmod N$

Then $U_j (j = 1, 2, \cdots, l)$ releases A_{lmj} to other cooperators in W.

Step 2 Any other cooperators in W obtain U_j's Y_{lj} and e_{lm} from the NB to verify the validity of A_{lmj}

$$(A_{lmj})^{e_{lm}} = Y_{lj} \qquad (2)$$

If Eq. (2) does not hold, they can ask U_j to release his/her valid sub-shadow.

Step 3 If all A_{lmj} released by the l participants in W are valid, every participant in W obtains H_{lm} from the NB and computes：

$$S'_{lm} = H_{lm} \oplus h\left[\left(\prod_{U_j \in W} (A_{lmj})^{\Delta_j} \bmod N \right) \right] \qquad (3)$$

Where $\Delta_j = \left[\prod_{U_i \in W, U_i \ne U_j} (-ID_i) / (ID_j - ID_i) \right]$;

And $S'_{lm} = S_{lm}$

Then all the secrets $(S_{i1}, S_{i2}, \cdots, S_{ik} \in S$ $1 \le i \le n, k = 1, 2, \cdots)$ can be reconstructed by performing this phase repetitively.

In the rest of the section, we will demonstrate the validity of the Eq. (1) , (2) and (3).

In the shadow generation and verification stage, any participant $U_j \in G$ can verify the correctness of his/her shadow released by U_D through the Eq. (1) as follows.

Participant U_j obtains $\{K_{ij}, g^{a_i}, g^{d_1}, \cdots, g^{d_{i-1}}\}$ $(i = 1, 2, \cdots, n)$ and computes: $x_{ij} = x_j \oplus K_{ij}$ $(i = 1, 2, \cdots, n)$,

because $K_{ij} = x_j \oplus C_{ij}$ $(i = 1, 2, \cdots, k)$,

so $x_{ij} = C_{ij}$. And $C_{ij} = f_i(ID_j) \bmod R (i = 1, 2, \cdots, k)$ then $x_{ij} = f_i(ID_j) \bmod R (i = 1, 2, \cdots, k)$, so we can demonstrate

$g^{x_{ij}} \bmod N$

$= g^{f_i(ID_j)} \bmod N$

$= g^{a_i + d_1(ID_j) + d_2(ID_j)^2 + \cdots + d_{i-1}(ID_j)^{i-1}} \bmod N$

$= g^{a_i} \prod_{m=1}^{i-1} (g^{d_m})^{(ID_i)^m} \bmod N$ $(i = 1, 2, \cdots, n)$

In the sub-shadow verification and secret reconstruction stage, any cooperator can verify the sub-shadow released by any $U_j \in W$ through the Eq. (2).

Assumed $U_j \in W$ is an honest participant. He/she uses his shadow computes his/her valid sub-shadow A_{lmj}, then the other cooperator can verify the sub-shadow A_{lmj} as follows.

$$\begin{aligned}
(A_{lmj})^{e_{lm}} &= \left[(g^{d_{lm}} \bmod N)^{K_{lj} \oplus x_j} \bmod N \right]^{e_{lm}} \\
&= \left[(g^{d_{lm}})^{x_j \oplus C_{lj} \oplus x_j} \bmod N \right]^{e_{lm}} \\
&= \left[(g^{d_{lm}})^{C_{lj}} \bmod N \right]^{e_{lm}} = \left[(g^{d_{lm}})^{C_{lj}} \bmod N \right]^{e_{lm}} \\
&= g^{C_{lj}} \bmod N = Y_{lj}
\end{aligned}$$

In the sub-shadow verification and secret reconstruction stage, every participant $U_j \in W$ can reconstruct the group secret $S_{lm} \in S$ through Eq.(3) as follows.

Assumed all A_{lmj} released by cooperators in W are valid, by the equations $H_{lm} = S_{lm} \oplus h(g^{a_l d_{lm}}) \bmod N$, $A_{lmj} = (g^{d_{lm}})^{C_{lj}} \bmod N$ $j = 1, 2, \cdots, l$, and $\Delta_j = \left[\prod_{U_i \in W, U_i \ne U_j} (-ID_i) / (ID_j - ID_i) \right]$, every participant can computes:

$$S'_{lm} = H_{lm} \oplus h\left[\prod_{U_j \in W} (A_{lmj})^{\Delta_j} \bmod N \right]$$

$$= S_{lm} \oplus h(g^{a_l d_{lm}}) \bmod N \oplus h\left[\left(\prod_{U_j \in W} (g^{d_{lm} C_{lj}}) \right)^{\Delta_j} \bmod N \right]$$

$$= S_{lm} \oplus h(g^{a_l d_{lm}}) \bmod N \oplus h(g^{f_l(0) d_{lm}} \bmod N)$$

$$= S_{lm} \oplus h(g^{a_l d_{lm}}) \bmod N \oplus h(g^{a_l d_{lm}} \bmod N)$$

$$= S_{lm}$$

III. ANALYSIS OF OUR SCHEME

A. Security analysis

In this section, we will analyze the security of our scheme. The security of our scheme is based on the intractability of the factorization and the discrete logarithm modulo a large composite problem. We will demonstrate that our scheme can withstand some possible attacks.

Attack 1 Some attackers may try to reveal the participants' secret shadows x_j from the public information in the NB. Attacker wants to reveal the participants' secret shadow x_j, he/she must firstly get the C_{ij}, then through the equation $K_{ij} = x_j \oplus C_{ij}$ $(i = 1,2,\cdots,n; j = 1,2,\cdots,n)$ and K_{ij} which is in the NB to get x_j.

Attack a Known the equations $Y_{ij} = g^{C_{ij}} \bmod N$, $(i = 1,2,\cdots,n; j = 1,2,\cdots,n)$. U_j's public key $Y_{ij}(i = 1,2,\cdots,n; j = 1,2,\cdots,n)$ and the parameters $\{N, g\}$, the difficult of getting C_{ij} is the same as breaking the discrete logarithm module a composite (DLMC) problem.

Attack b Know the equations

$A_{ikj} = (T_{ik})^{B_{ij}} = (g^{d_{ik}} \bmod N)^{C_{ij}} \bmod N$,

$(i = 1,2,\cdots,n; j = 1,2,\cdots,n; k = 1,2,\cdots)$

$T_{ik}(i = 1,2,\cdots,n; k = 1,2,\cdots)$ and the parameters $\{N, g\}$, as the attack (a), the adversary also face the difficult of breaking the discrete logarithm module a composite (DLMC) problem.

Attack 2 In the sub-shadow verification and secret reconstruction stage, a malicious participant may try to send a false sub-shadow A'_{lmj} satisfying the Eq. (2) to other participants, so that only he/she can construct the real group secret S_{lm}.

The malicious participant should first modify his/her public key Y_{lj} and the U_D's public key e_{lm} in the NB to compute A'_{lmj} satisfying the Eq. (2). However, to modify the e_{lm} to e'_{lm}, the malicious participant should first compute the d'_{lm} corresponding e'_{lm}, this is as difficult as breaking the RSA public cryptography. Furthermore, the contents of the NB can only be modified or updated by U_D. Thus, the dishonest participant cannot release a fake sub-shadow A'_{lmj} to pass Eq. (2).

Attack 3 A malicious participant in W may use A_{lmj} $(j = 1,2,\cdots,l)$ to construct the remaining secrets in S after constructing the secret S_{lm} with the other cooperators in W.

Attack a After constructing the secret S_{lm}, the malicious participant may use the l sub-shadows A_{lmj} $(j = 1,2,\cdots,l)$ to construct the remaining secrets corresponding the threshold value l in S such as S_{ln} by the Eq. (3). However, the malicious participant must break the RSA public cryptography to obtain the d_{ln}, then computes the $g^{a_i d_{ln}} \bmod N$ to get S_{ln}. It is not possible.

Attack b After constructing the secret S_{lm}, the malicious participant may use the l sub-shadows A_{lmj} $(j = 1,2,\cdots,l)$ to construct the remaining secrets corresponding to the threshold value $l'(l' < l)$ in S such as $S_{l'm}$ by the Eq. (3). The malicious participant will face the same difficulty as the attack 3 (a).

B. Dynamic analysis

On the practical application of secret sharing scheme, there are some problems to meet, such as adding new participant and group secret, U_D must make a cost of calculation and communication to renew the shadows of the old participants. Our scheme not only can effectively solve the above problems but share the group secrets according to the different threshold values, even every threshold value corresponding many group secrets.

If U_D wants to add new group secret S_{new} for sharing in the system, U_D first decides the threshold value for the group secret S_{new}, then performs the credit ticket generation stage to create $(e_{new}, T_{new}, H_{new})$ and puts on the NB. However, the shadows of the participants remain unchanged.

If U_D wants to add a new participant in the system, U_D only performs the shadow generation and verification stage to send shadow x_{new} to the new participant and put K_{inew} $i = 1,2,\cdots,n$ on the NB. The old participants' shadows remain unchanged.

C. Performance analysis

In this section, we firstly compare the system performances of our scheme with the current multi-policy secret sharing schemes, and then compare the verifiable complexity of our scheme with the current verifiable multi-secret sharing schemes.

Table 1 Comparison among our scheme, Geng et al.'s scheme and Lin et al's scheme

	Geng et al's scheme	Lin et al's scheme	Our scheme
System initialization complexity	0	0	0
Shadow generation complexity	$k(2n+1)T_e$ $+knT_m$	knT_m	knT_e $+k(n+2)T_m$
Group secret reconstruction complexity	$2kT_e$ $+(2k^2-k)T_m$	$2(k^2-k)T_m$	$2kT_e$ $+(k^2-k)T_m$
The initial number of group secrets for sharing	n	n	nk
Against cheating by U_D	NO	NO	YES
Against cheating by participants	NO	NO	YES
Shadows' reusability	YES	YES	YES

Group secrets corresponding threshold value	YES	YES	YES
Multi-secret sharing corresponding every threshold	NO	NO	YES
Add new participant	NO	NO	YES
Add new group secret	NO	NO	YES

T_m : the time for performing a modular multiplication computation.

T_e : the time for performing a modular exponentiation computation

In table 1, we compare the computation complexity of system and functions among our scheme, Geng et al's scheme and Lin et al's scheme. As the result shown in Table 1, they are all multi-policy secret sharing schemes, but our scheme's initial number of group secrets for sharing is larger than the other schemes because our scheme can share many group secrets according to different threshold value. The computation complexity of our scheme is lower than the Geng et al's scheme and higher than Lin et al's scheme, however, our scheme can realize dynamicity such as add new participants and new group secrets. Moreover, our scheme can detect the malicious secret dealer and dishonest participants. Generally speaking, our scheme outperforms Geng et al.'s scheme and Lin et al's scheme.

Table 2 Comparison among our scheme, Chen et al's scheme and Chang et al's scheme

	Chen et al's scheme	**Chang et al's scheme**	**Our scheme**
Against cheating by U_D complexity	$2tT_e$	$2tT_e$	$2tT_e$
Against cheating by participants complexity	$(t-1)T_e$	$2(t-1)T_e$	$(t-1)T_e$
The number of public values for reconstructing one group secret	$n+2$	3	3
The initial number of group secrets for sharing	k	k	nk
Shadows' reusability	NO	YES	YES
Group secrets corresponding threshold value	NO	NO	YES
Multi-secret sharing corresponding every threshold	NO	NO	YES
Add new participant	YES	NO	YES
Add new group secret	NO	YES	YES

T_e : the time for performing a modular exponentiation computation

In table 2, we mainly compare the computation complexity of verifiability and functions among our scheme, Chen et al's scheme and Chang et al's scheme. As the result shown in Table 2, the verifiable computation complexity of our scheme is the lowest, the initial number of group secrets for sharing is the greatest and the amount of public information published by the dealer for reconstructing a secret required is less than that required by the Chen et al. scheme. Moreover, our scheme is multi-policy secret sharing scheme that can share group secrets according to different threshold values, even share many group secrets corresponding to every threshold value. Those are the other two schemes can not realize. So our scheme outperforms Chen et al's scheme and Chang et al's scheme.

IV. CONCLUSION

In this paper, we review the current secret sharing schemes and point out their problems. Taking all these problems into consideration, we propose a novel verifiable dynamic multi-policy secret sharing scheme based on the intractability of the factorization and the discrete logarithm modulo a large composite problem.

Compared to Geng et al's and Lin et al's multi-policy secret sharing scheme, our proposed scheme can share many group secrets according to different threshold value, add new participants and new group secrets, detect the malicious secret dealer and dishonest participants. These are Geng et al's and Lin et al's multi-policy secret sharing scheme can not achieve. Compared to Chen et al's and Chang et al's verifiable multi-secret sharing scheme, our proposed scheme can share group secrets according to different threshold values, even share many group secrets corresponding to every threshold value. Those are the other two schemes can not realize. Our proposed scheme's verifiability also outperforms their schemes. In general, our proposed scheme outperforms the current secret sharing schemes and has more functions.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their useful comments. This work is supported by National Basic Research Program of China (973 Program) (No. 2007CB310704), National Natural Science Foundation of China（No. 90718001, 60821001），The 111 Project (No. B08004) and National 863 (No. 2008AA011004)

REFERENCES

[1] I. Ray, Narasimhamurthi N. "An anonymous electronic voting protocol for voting over the Internet," *Third International Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems*, Los Alamitos, Calif: IEEE Computer Society, pp. 188-190, 2001.

[2] P. A. Fouque, G, Poupard, J. Stern. "Sharing decryption in the context of voting or lotteries," *Proceeding of Financial Cryptography 2000*, Berlin, Germany: Springer Verlag, 2000, pp. 90-94.

[3] A. Shamir. "How to share a secret," *Communication of the ACM*, vol. 22, pp. 612-613, 1979.

[4] G. Blakley. "Safeguarding cryptographic keys,"*Proc AFIPS 1979 Nalt Conf*, New York: AFIPS Press, pp. 313-317, 1979.

[5] J. He, E. Dawson. "Multistage secret sharing based on one-way function," *Electron. Lett.* Vol.30, pp. 1591-1592, 1994.

[6] L. J. Pang, H. X. Li, Y. Yao, Y. M. Wang. "A Verifiable (t,n) Multiple Secret Sharing Scheme and Its Analyses," *2008 International Symposium on Electronic Commerce and Security*, Guangzhou, China: IEEE Computer Society, pp. 22-26, 2008.

[7] M. Tompa, H. Woll. "How to share a secret with cheaters," *J. Crypto*, vol.1, pp.133–138, 1988.

[8] J. Ao, G. S. Liao, C. B. Ma. "A Novel Non-interactive Verifiable Secret Sharing Scheme," *10th International Conference on Communication Technology*, Guilin, China: IEEE Computer Society, pp.1-4, 2006.

[9] T.Y. Chang, M.S. Hwang, W.P. Yang. "An improvement on the Lin-Wu (t,n) -threshold verifiable multi-secret sharing scheme," *Applied Mathematics and Computation*, vol.163, pp.169-178, 2005.

[10] P. Dong. X H. Kuang. X. C. Lu. "A non-interactive protocol for member expansion in secret sharing scheme," *Journal of Software*, vol.16, pp.116-120, 2005.

[11] Y. J. Geng, X. H. Fan, F. Hong. "A New Multi-secret Sharing Scheme," *9th International Conference on Advanced Communication Technology*, Phoenix Park, Korea: National Computerization Agency, pp.1515-1517, 2005.

[12] H. Y. Lin, Y. S. Yeh. "Dynamic Multi-Secret Sharing Scheme," *International Journal of Contemporary Mathematical Sciences*, vol.3, pp. 37 – 42, 2008.

[13] J. Shao, Z. F. Cao. "A new efficient (t, n) verifiable multi-secret sharing (VMSS) based on YCH scheme," *Applied Mathematics and Computation*, vol. 168, pp. 135-140, 2005.

[14] H. Y. Chien, J. K. Jan, Y. M. Tseng. "A practical (t,n) multi-secret sharing scheme," *IEICE Trans. Fundamentals*, E83-A (12), pp.2762–2765, 2000.

[15] T. P. Pederson. "Distributed Prover with Applications to Undeniable Signatures," *In Advances in Cryptology proceedings of EUROCRYPT'91,* Lecture Notes in Computer Science. Berlin, Germany: Springer, 1992, pp. 221-242.

[16] T. Y. Chang, M. S. Hwang, W. P. Yang. "A new multi-stage secret sharing scheme using one-way function," *Association for Computing Machinery*, vol.39, pp.48-55, 2005.

[17] W. A. Jackson, K. M. Martin, C. M. O'Keefe. "On sharing many secrets," *Asiacrypt'94*, pp: 42-54, 1994.

[18] L. Harn. "Efficient sharing (broadcasting) of multi –secret," *IEE Proceedings Computers and Digital Techniques*, vol.142, pp.237-240, 1995.

[19] T.Y. Lin, T.C. Wu. "(t,n) threshold verifiable multi -secret sharing scheme based on factorization intractability and discrete logarithm module a composite problems," *IEE Proceedings Computers and Digital Techniques*, vol.146, pp.264-268, 1999.

[20] L. Chen, D. Gollmann, C.J. Mitchell, P. Wild. "Secret sharing with reusable polynomials," *Proceedings of ACISP '97*, pp. 183–193, 1997.

[21] J. J Zhao, J. Z. Zhang, R. Zhao. "A practical verifiable multi-secret sharing scheme," *Computer Standards & Interfaces*, vol. 29, pp.138-141, 2007.

[22] B. Chor, S. Goldwasser, S. Micali, et al. "Verifiable secret sharing and achieving simultaneity," *In the presence of faults: proceedings of the 26th IEEE Symposium on the Foundations of Computer Science (FOCS)*. Washington: IEEE Computer Society, 1985.

[23] M. Stadler. "Public verifiable secret sharing," *Proceeding of Advances in Cryptology -Eurocrypt'96*, Berlin, Germany: Springer-Verlag, 1996.

[24] S. Bruce. *Applied Cryptography: protocols, algorithms, and source code in C* 2nd Edition. New York: Springer-Verlag, pp. 334 -339, 1996.